

Package ‘RSQLServer’

June 17, 2017

Type Package

Title SQL Server R Database Interface (DBI) and 'dplyr' SQL Backend

Version 0.3.0

Description Utilises The 'jTDS' project's 'JDBC' 3.0 'SQL Server' driver to extend 'DBI' classes and methods. The package also implements a 'SQL' backend to the 'dplyr' package.

License GPL-2

Depends R (>= 3.2.0)

Imports methods (>= 3.2.0), DBI (>= 0.5.0), rJava (>= 0.9.6), dplyr (>= 0.7.0), dbplyr (>= 1.0.0), assertthat (>= 0.1), yaml (>= 2.1.13), purrr (>= 0.2.1)

Suggests testthat, DBItest (>= 1.4), covr

SystemRequirements Java (>= 1.3)

URL <https://github.com/manuelcostigan/RSQLServer>

BugReports <https://github.com/manuelcostigan/RSQLServer/issues>

Collate 'RSQLServer.R' 'Utils.R' 'dbi-classes.R' 'dbi-methods.R' 'dbi-shims.R' 'dbi-tables.R' 'dplyr-imports.R' 'java-shims.R' 'jdbc-methods.R' 'onLoad.R' 'sql-backends.R'

RoxygenNote 6.0.1

NeedsCompilation no

Author Imanuel Costigan [aut, cre],
The jTDS Project (for MSSQL Server driver) [aut],
Simon Urbanek [ctb],
Hadley Wickham [ctb],
Hong Ooi [ctb],
Romain Francois [ctb],
Bill Evans [ctb],
RStudio [cph],
The Legion Of The Bouncy Castle [cph, ctb]

Maintainer Imanuel Costigan <i.costigan@me.com>

Repository CRAN

Date/Publication 2017-06-17 14:33:29 UTC

R topics documented:

have_test_server	2
RSQLServer	3
SQLServer	3
Index	5

have_test_server	<i>Checks availability of TEST server</i>
------------------	---

Description

SQL Server details can be specified in a `~/sql.yaml` file. To be able to run examples and some tests in the package, it is necessary for there to be a valid server with name `TEST` in this file.

Usage

```
have_test_server(type = "sqlserver")
```

Arguments

`type` specifies whether the server type is "sqlserver" (default) or "sybase"

Value

boolean TRUE if TEST server details are available. Otherwise, FALSE

See Also

[get_server_details dbConnect, SQLServerDriver-method](#)

Examples

```
have_test_server()
```

RSQLServer

*RSQLServer***Description**

Implements an R Database Interface (DBI) for SQL Server. Sybase may work, but to date has not been tested. You will need to have installed the [Java Runtime Environment](#).

Details

If you intend to use integrated security (Windows Authentication) to authenticate your server session, you will need to download [jTDS](#) and copy the native single sign on library (ntlmauth.dll) to any location on your system's PATH (e.g. `Sys.getenv("PATH")`). Full installation instructions are available in the README.SSO file in the jTDS download bundle. This functionality is unreliable in my testing, but it could just be my setup. I would recommend that that you fully specify your server and login credentials in the `~/sql.yaml` file rather than using the single sign on library. See the example provided: `system.file("extdata", "sql.yaml", package = "RSQLServer")`

SQLServer

*SQLServerDriver class and methods***Description**

`SQLServer()` creates a `SQLServerDriver` object and is based on the jTDS driver while `dbConnect()` provides a convenient interface to connecting to a SQL Server database using this driver.

Usage

```
SQLServer()
```

```
## S4 method for signature 'SQLServerDriver'
dbConnect(drv, server, file = NULL,
          database = NULL, type = NULL, port = NULL, properties = NULL)
```

Arguments

<code>drv</code>	An object of class SQLServerDriver , or an existing SQLServerConnection . If a connection, the connection will be cloned.
<code>server</code>	the server address or recognised alias thereof.
<code>file</code>	defaults to using the server details file in <code>\$HOME/sql.yaml</code> . The server details including type, port and any optional properties can be sourced from this file. If the server name is found in file, the details therein are used (and in particular, those provided in other arguments to this function are ignored). The connection method prefers server details to be provided in a "sql.yaml" file rather than provided as arguments to this function. If you wish to specify the details as parameters, ensure that either the file does not exist or that the server details are not in the YAML file.

database	the name of the database hosted on the server. If an empty string or NULL (default), a connection to the default database on server is assumed.
type	the server type. Must be either "sqlserver" or "sybase". Defaults to "sqlserver" when set to NULL.
port	the TCP/IP default port. This will be coerced to a string. Defaults to 1433 if set to NULL or an empty string (jTDS behaviour).
properties	One or more optional connection properties , in a named list (defaults to empty list). Note if you intend to set the useNTLMv2 property to 'true' from the default API value of 'false', you will need to make a specific authentication driver available to the SQL Server driver, although this has not worked particularly well in testing. See RSQLServer for more details. Should you wish to use Windows authentication to connect to the server, I recommend you set the following optional parameters: set useNTLMv2 to 'true', domain to your domain and user and password to your username and password on domain. jTDS' SSO functionality is flaky.

Value

SQLServer() returns an object of class SQLServerDriver; dbConnect() returns a [SQLServerConnection](#) object.

References

[jTDS API doc for Driver class](#)

Examples

```
## Not run:
SQLServer()

## End(Not run)
# View sql.yaml file bundled in package
file <- system.file("extdata", "sql.yaml", package = "RSQLServer")
readLines(file)
# Connect using ~/sql.yaml file
## Not run:
if (have_test_server()) {
  dbConnect(RSQLServer::SQLServer(), "TEST")
}
# Example where ~/sql.yaml does not exist or where the server
# is not in the YAML file.
dbConnect(RSQLServer::SQLServer(), server="11.1.111.11", port=1434,
  properties=list(useNTLMv2="true", domain="myco", user="me",
    password="asecret"))

## End(Not run)
```

Index

dbConnect, [SQLServerDriver-method \(SQLServer\)](#), [3](#)

get_server_details, [2](#)

have_test_server, [2](#)

RSQLServer, [3](#), [4](#)

RSQLServer-package ([RSQLServer](#)), [3](#)

SQLServer, [3](#)

SQLServerConnection, [3](#), [4](#)

SQLServerDriver, [3](#)